IN THE CLAIMS

Please amend the claims as follows.

1. (Previously Presented) A method for using a binary state machine for processing a data stream in an intrusion detection system, the method comprising:

maintaining a state table, the state table indexed such that inputs comprising a current state and a current character yield an output of a new state, the new state related to an indication of an attack on a computer network;

maintaining the current state;

receiving, at a state machine of an intrusion detection device, an input stream destined for a first network device to be protected by the intrusion detection device, ~~at a binary state machine prior to being buffered at a first network device~~, the input stream received at the state machine prior to reaching the first network device and comprising a plurality of characters, wherein the first network device is operable to execute a program ~~transmitted by a second network device~~;

~~storing a copy of the input stream at a network interface disposed between the first network device and the second network device;~~

~~repeating the following for each character of the plurality of characters;~~

selecting a first character of the input stream as the current character; and

comparing ~~a current character~~ the current character and the current state to the state table to generate a new state~~; and~~

~~discarding the first character before selecting a next character of the input stream; and~~

~~transmitting the copy of the input stream to the first network device if an attack on the computer network is not detected~~.

2. (Original) The method of Claim 1, further comprising initializing the current state to an initial state.

3. (Original) The method of Claim 1, further comprising:

setting the current state equal to the new state;

selecting a next character as the current character, the next character appearing subsequent to the first character in the input stream; and

repeating the comparing step.

4. (Original) The method of Claim 1, further comprising recognizing the new state as indicative of an attack upon the computer network.

5. (Original) The method of Claim 5, further comprising sounding an alarm.

6. (Original) The method of Claim 1, further comprising generating the state table from a REGEX command.

7. (Currently Amended) A system for use as a binary state machine for processing a data stream in an intrusion detection system, the system comprising:

a state table indexed such that inputs comprising a current state and a current character yield an output of a new state, the new state related to an attack on a computer network; and

a state machine communicatively coupled to the state table, the state machine operable to:

maintain the current state;

receive an input stream destined for a first network device to be protected by the intrusion detection system, ~~prior to being buffered at a first network device~~, the input stream received prior to reaching the first network device and comprising a plurality of characters, wherein the first network device is operable to execute a program ~~transmitted by a second network device~~;

~~repeat the following for each character of the plurality of characters;~~

select a first character of the input stream as the current character; and

compare ~~a current character~~ the current character and the current state to the state table to generate a new state~~; and~~

~~discard the first character before selecting a next character of the input stream; and~~

~~a network interface disposed between the first network device and the second network device and operable to:~~

~~store a copy of the input stream; and~~

~~transmit the copy of the input stream to the first network device if an attack on the computer network is not detected~~.

8. (Original) The system of Claim 7 further comprising a computer readable medium, wherein the state table is stored upon the computer readable medium.

9. (Original) The system of Claim 8, wherein the state machine comprises software code stored upon the computer readable medium, the software code further operable to be executed by a computer processor.

10. (Original) The system of Claim 7, wherein the state machine is further operable to initialize the current state to an initial state.

11. (Currently Amended) The system of Claim 7, wherein the state machine is further operable to:

~~setting~~ set the current state equal to the new state;

~~selecting~~ select a next character as the current character, the next character appearing subsequent to the first character in the input stream; and

~~repeating~~ repeat the comparing step.

12. (Original) The system of Claim 7, wherein the state machine is further operable to recognizing the new state as indicative of an attack upon the computer network.

13. (Currently Amended) A system for use as an intrusion detection system, the system comprising:

a computer readable medium;

a network interface for receiving an input stream destined for a first network device to be protected by the intrusion detection system, ~~prior to being buffered at a first network device,~~ the network interface operable to receive the input stream before the input stream reaches the first network device, the input stream comprising a plurality of characters transmitted by a second network device, wherein the first network device is operable to execute a program;

a processor communicatively coupled to the computer readable medium and the network interface;

a state table stored upon the computer readable medium, the state table indexed such that inputs comprising a current state and a current character yield an output of a new state, the new state related to an attack on a computer network; and

a state machine comprising instructions stored upon the computer readable medium and executable by the processor, the state machine communicatively coupled to the state table, the state machine operable to:

maintain the current state;

~~repeat the following for each character of the plurality of characters;~~

select a first character of the input stream as the current character; and

compare ~~a current character~~ the current character and the current state to the state table to generate a new state~~; and~~

~~discard the first character before selecting a next character of the input stream, the network interface further operable to:~~

~~store a copy of the input stream; and~~

~~transmit the copy of the input stream to the first network device if an attack on the computer network is not detected~~.

14. (Currently Amended) A logic for using a binary state machine for processing a data stream in an intrusion detection system, the logic embodied in a computer-readable medium and operable to:

maintain a state table, the state table indexed such that inputs comprising a current state and a current character yield an output of a new state, the new state related to an indication of an attack on a computer network;

maintain the current state;

receive an input stream destined for a first network device to be protected by the intrusion detection system, ~~at a binary state machine prior to being buffered at a first network device,~~ the input stream received at the logic prior to reaching the first network device and comprising a plurality of characters, wherein the first network device is operable to make a decision according to a program ~~transmitted by a second network device~~;

~~store a copy of the input stream at a network interface disposed between the first network device and the second network device;~~

~~repeat the following for each character of the plurality of characters;~~

select a first character of the input stream as the current character; and

compare ~~a current character~~ the current character and the current state to the state table to generate a new state.~~; and~~

~~discard the first character before selecting a next character of the input stream; and~~

~~transmit the copy of the input stream to the first network device if an attack on the computer network is not detected~~

15. (Previously Presented) The logic of Claim 14, further operable to initialize the current state to an initial state.

16. (Previously Presented) The logic of Claim 14, further operable to:

set the current state equal to the new state;

select a next character as the current character, the next character appearing subsequent to the first character in the input stream; and

repeat the comparing step.

17. (Previously Presented) The logic of Claim 14, further operable to recognize the new state as indicative of an attack upon the computer network.

18. (Canceled).

19. (Previously Presented) The logic of Claim 14, further operable to generate the state table from a REGEX command.

20. (Previously Presented) An intrusion detection system, comprising:

means for maintaining a state table, the state table indexed such that inputs comprising a current state and a current character yield an output of a new state, the new state related to an indication of an attack on a computer network;

means for maintaining the current state;

means for receiving an input stream destined for a first network device to be protected by the intrusion detection system, ~~prior to being buffered at a first network device,~~ the input stream received at the means for receiving the input stream prior to reaching the first network device and comprising a plurality of characters, wherein the first network device is operable to execute a program ~~transmitted by a second network device~~;

~~means for storing a copy of the input stream, the means for storing disposed between the first network device and the second network device;~~

means for selecting a first character of the input stream as the current character; and

means for comparing ~~a current character~~ the current character and the current state to the state table to generate a new state~~; and~~

~~means for discarding the first character before selecting a next character of the input stream; and~~

means for transmitting the copy of the input stream to the first network device if an attack on the computer network is not detected.

21. The method of Claim 1, and further comprising:

setting the current state equal to the new state;

selecting a next character as the current character, the next character appearing subsequent to the first character in the input stream;

repeating the comparing step; and

wherein the first character and the next character are each selected and compared only once.